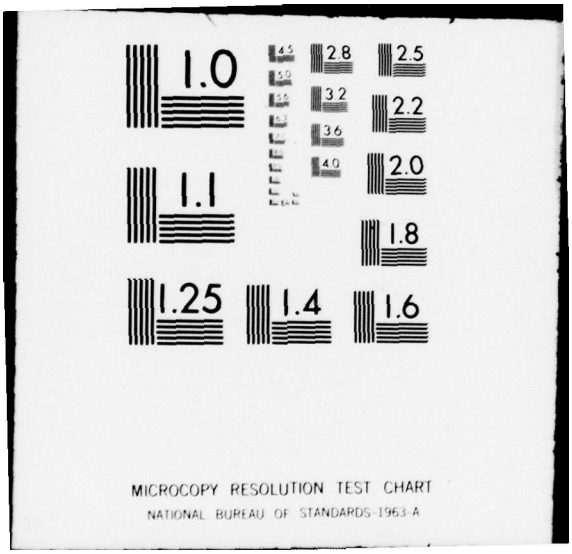
AD-A059 632 UNIVERSITY COLL LONDON (ENGLAND) DEPT OF STATISTICS --ETC F/G 17/2
THE MEASURED CHARACTERISTICS OF TRAFFIC IN THE UCL NODE OF THE --ETC(U)
JAN 77 S TREADWELL
UNCLASSIFIED TR-33 N00014-77-G-0005
NL

END
DATE FILMED
12-78
DDC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER (14) TR-C33	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) (6) The Measured Characteristics of Traffic in the UCL Node of the ARPA Network,	5. TYPE OF REPORT & PERIOD COVERED Technical, 1977
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) (10) Steve Treadwell	8. CONTRACT OR GRANT NUMBER(s) (15) N-00014-77-G-0003
9. PERFORMING ORGANIZATION NAME AND ADDRESS University College London, Gower Street, London WC1, ENGLAND	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Defence Advanced Research Projects Agency (11) 1400 Wilson Boulevard, Arlington, Virginia 22209, USA	12. REPORT DATE January 1977
	13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Office of Naval Research (12) 29 p.	15. SECURITY CLASS. (of this report) UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE U

16. DISTRIBUTION STATEMENT (of this Report)

No.1 This document has been approved for public release and sale; its distribution is unlimited. (9) Technical Repts

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Measurements of Network Usage, Packet Switched Networks, ARPANET and Access Control

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A novel approach to network traffic analysis is presented. Information on the low-level line traffic is recorded on a magnetic tape. Offline analysis of this data enables us to reconstruct high-level protocols in use in the ARPANET. Accurate timings of the low level traffic allows us to do detailed analysis of performance experienced by network users.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE 409 724

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

CONTENTS

1. Introduction

This paper describes a method of recording the characteristics of traffic passing between two nodes of the ARPA network, and the offline analysis of this data, including some preliminary results. The recording of information by this method and the later analysis has the important advantage over other measurement methods of not interferring with the performance of the network in any way.

The measurement was effected on the communication line linking the UCL node (or IMP) with the rest of the ARPA network. As the UCL node is 'spur', i.e. only connected by this one line, the measurement tool, when operating, can measure the characteristics of all traffic flowing to or from the UCL node and the ARPA network.

The UCL node is purely experimental, and any data aquisition is only for research purposes. All UK users of the UCL node are informed that data aquisition may occur; it has been approved specifically by the governing committee of the project. Nevertheless, the data aquisition is done very rarely and for specific network performance investigations. The data contents are never recorded or analysed.

2. Data Aquisition

2.1 IMP -IMP Communications

The IMPs communicate with each other over standard Post Office data communication lines. The lines between IMPs are always in pairs, one line for traffic traveling in each direction. This enables the IMPs to transmit and receive data at the same time. A schematic of this configuration is shown in Fig. 1.

Although traffic is sent down the communication lines a bit at a time, the data is usually considered in logical units of eight bits called bytes of characters. The number of characters that can be sent down the line in a given period of time is determined directly from the baud rate (bits per second capacity) of the line. In the example cited in this paper, the traffic measured was travelling between the London IMP and an IMP in Norway (IMPs 42 and 41 respectively) which communicate on a 9.6K baud line or 1,200 characters per second.

The IMP hardware expects a continuous stream of characters to be sent down each line at the specified baud rate of the line.

However, the IMP software usually requires to send data at that rate for short intervals; the rest of the time when there is no data, the hardware sends special dummy characters called synchronising characters. These characters ensure that the hardware at the receiving IMP keeps the logical grouping of bits into bytes synchronised with the hardware of the transmitting IMP.

Related data bytes that are all going to the same destination are grouped together to form packets. Each packet has an address field, or header, affixed to the front of it to indicate the packet's network destination. The header is about a dozen bytes itself, and even if just one data byte is to be sent through the network it still has to be accompanied by the header. From this minimum packet size of 12 bytes the size of a packet can range up to about 128 bytes, including the header.

To distinguish the data in the packets from the synchronising characters, each packet is delimited by a pair of characters. A control character followed by a start-of-text character is placed at the beginning of the packet and a control character followed by the end-of-text character is placed at the end. There is of course the chance that the data in the packet may contain a bit pattern that matches one of these pairs. To avoid any confusion the transmitting hardware of the IMP doubles any control byte found in the data portion of the packet, i.e. replaces the control byte by two control bytes. The receiving hardware performs the opposite operation. Only an unpaired control byte is interpreted as being a real control byte.

Following immediately after each packet is a hardware generated checksum. This checksum occupies three bytes and is never seen by the IMP software. If the IMP receives a packet whose sumcheck is incorrect, the packet is automatically discarded.

Figure 2 illustrates the packet format described above.

2.2 Measuring the IMP Traffic

As is shown in Fig. 1, the traffic to and from our IMP was measured between the IMP and the modem. The data was collected here only for simplicity as between the IMP and the modem the data is in digital form and is more readily acceptable to our computer.

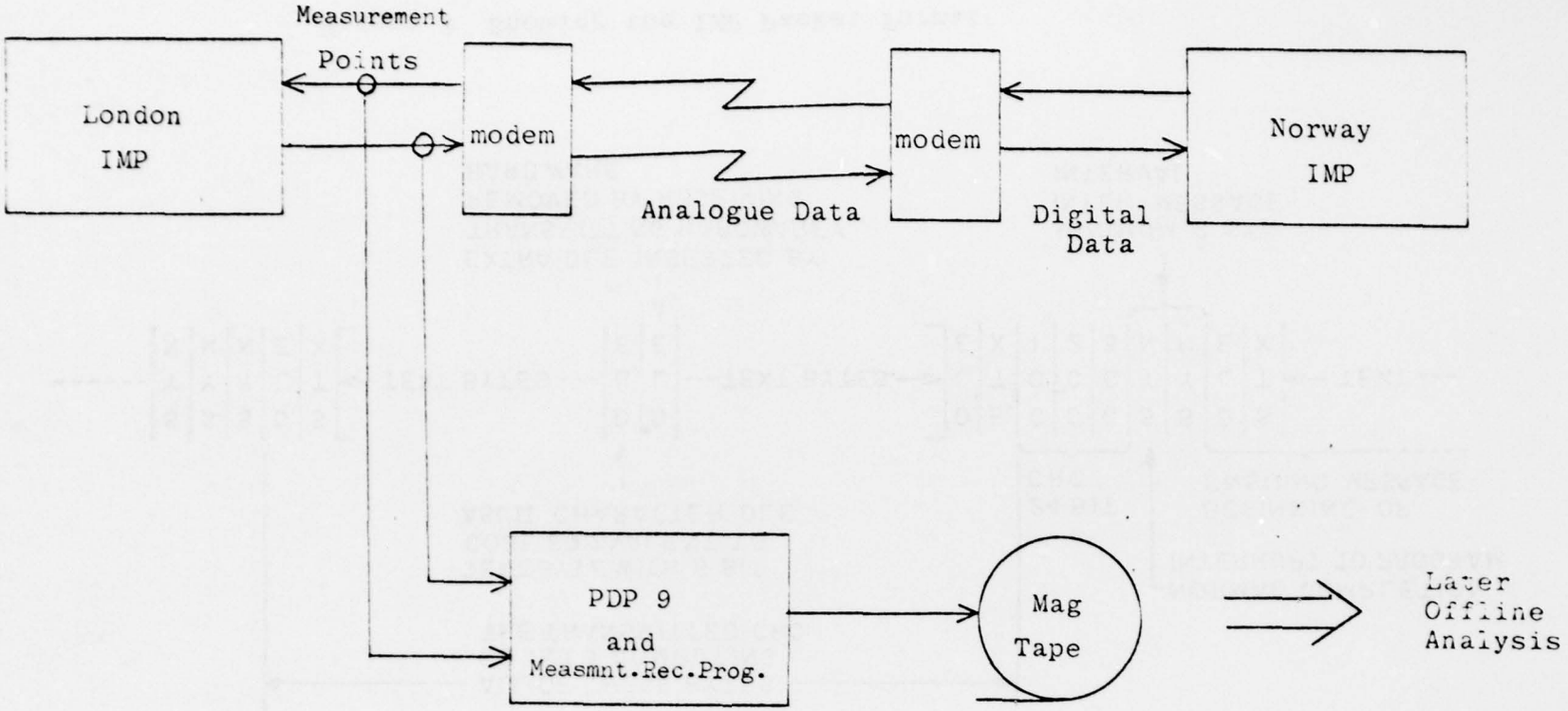


Figure 1 Showing the Measurement Configuration.

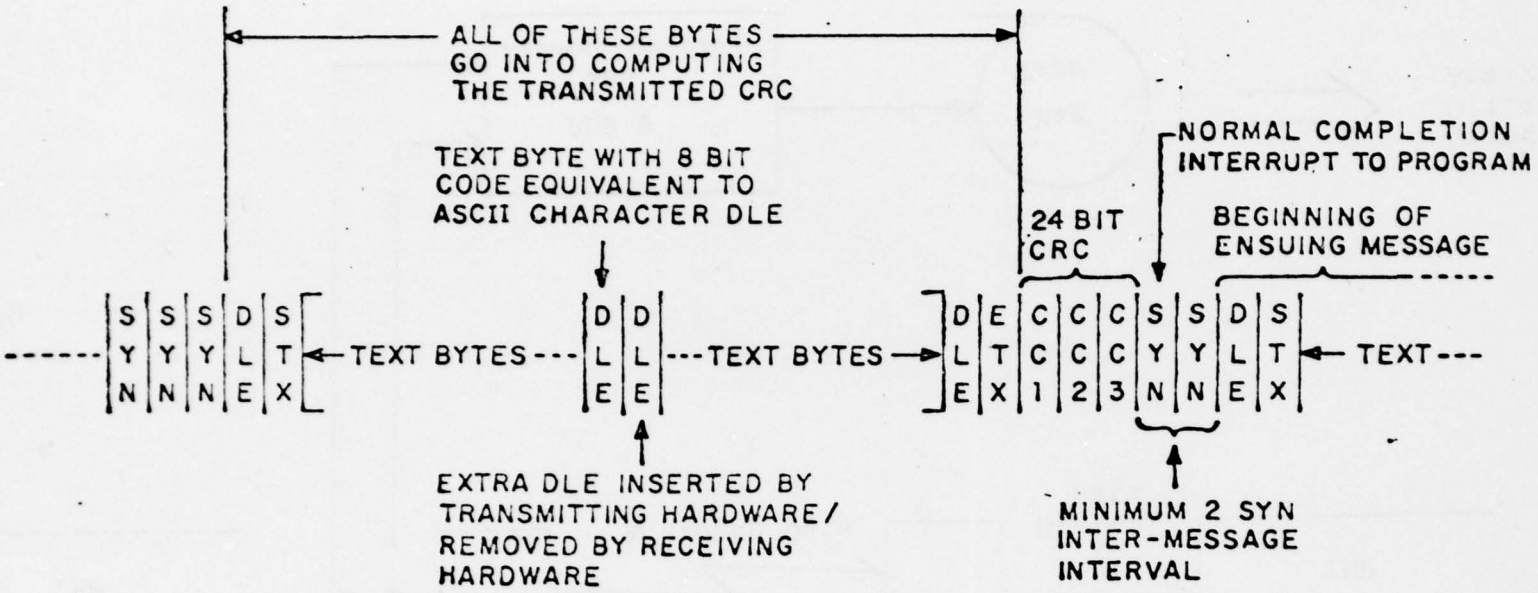


Figure 2 Showing the IMP Packet Format

Measuring traffic performance is this way has no affect upon the IMP or the modem and neither is aware that the traffic is actually being measured. The traffic passing between the IMP and the modem is not interferred with or delayed in any way. Indeed, the tap does not have a "write" ability at all, making it physically impossible to modify the data.

The data collection lines run to our PDP 9 where a program ignores all the synchronising characters and strips off the pair of characters at the beginning and end of each packet. Any doubled control characters in the packet are reduced to just one control character. The sumcheck is also read as part of the packet but no attempt is made by the program to verify it, due to the large CPU overheads that this would involve.

2.3 Recording the Measurement Data

In order to extract information about each packet, the PDP 9 has to receive each packet intact. As soon as the complete packet has been received, the measurement program quickly throws away the data portion of the packet, keeping only the header, control information and the length of the packet data area.

Although the capacity of each line is 1,200 characters per second, the rate at which data is sent on the average is far below this. However, occasionally there are bursts of data when several packets arrive in quick succession. Unfortunately, the PDP 9 CPU is not fast enough to be able to move a packet out of a buffer in the three milliseconds before another packet can be expected. This constraint, that there is not sufficient time to "move" a packet about in core, dictated the approach to recording the data.

As a packet cannot be moved once it has been read into core, it must be read directly into the tape output buffer. As writing to the tape is carried out in parallel to receiving the data on the input lines, this should enable the PDP 9 to keep up with the data flow.

There is however the further problem that there are two input lines that can both receive data at the same time. If data from one line is being received and placed into the tape output buffer when a second packet begins to arrive on the other line there is no place to put the second packet. The second packet cannot be placed after the packet currently being

read into the buffer as the length of that packet is unknown until the end of it is detected, and it cannot be read into another place in core as there is not enough time to move it into the buffer later.

To get around these problems, the measurement program keeps three buffers for each input line. These buffers are 2000 bytes long giving a total buffer space of 4000 bytes for each line (assuming that one buffer may be busy being written to tape). This allows about three seconds of data at peak data rate, to be recorded from each line, plenty of time for the tape to recover from any temporary tape errors.

As each buffer is filled, it is placed in the tape output buffer queue. When the tape has written the buffer it is returned to the appropriate line to be used again. As the data rate of the tape exceeds the combined data arrival rate of the input lines it can easily keep up to the data flow. However, the buffering technique described above has a serious consequence that is explained below.

2.4 Sorting the Recorded Data

Although the data arriving from both input lines is multiplexed to the tape unit it is multiplexed by buffer of one second worth of data from one line followed by a buffer of data from the other line for the same second. If the traffic on one line was far heavier than the other, it is quite possible to have several buffers from one line before one from the other.

For any reasonable analysis it is important to know the precise time at which a packet arrived. Both problems are solved by the measurement program placing a time stamp at the beginning of each packet. The time stamp is accurate to a hundredth of a second and indicates the time of arrival in the PDP 9 of the first byte of the packet.

This time stamp thus serves two purposes. It is used first of all to sort the data into the correct chronological sequence before it is analysed and it is used by the analysis program to enable accurate timings of data rates and protocols.

3. ARPA Network Protocols

In order to understand how the analysis program works and how it manages to analyse traffic of individual users in the network, it is necessary to be familiar with the protocols in the network.

There are two main levels of protocol existing in the ARPA network. At the low level is the IMP protocol which controls such things as routing in the network and flow control and at the higher level is the host protocol, which controls the actual user to user connections across the network.

3.1 IMP to IMP Protocol

The IMP level protocol falls into four main categories: data, control, routing and special. The data type (called type 0) actually carries the users data and the end-to-end acknowledgements. This type of traffic will be discussed in more detail below. The control traffic (type 1) is used to obtain buffer space in the source and destination IMPs for two hosts that are exchanging traffic across the network. Control traffic is sent as a direct result of hosts setting up connections and is closely tied in with type 0 traffic.

Routing traffic is sent out by neighbouring IMPs approximately every .6 seconds. This type 2 packet consists of a table with an entry for each IMP in the network indicating both the number of hops and the time taken to reach that IMP. All IMPs send their own personal view of the network to each of its neighbours which use it to update their own routing information. As well as sending the actual routing table the IMPs also exchange acknowledgements for the previous routing table. Into this same category fall "hello" and "I heard you" packets which are used to test if the line connecting a pair of IMPs is still operational.

Special (type 3) traffic is seldom seen on the network. Its main function is debugging and network status. For instance, should an IMP program collapse this type of traffic is used to reload the program. An IMP program can also detect that something is wrong and demand to be reloaded using type 3 traffic.

The IMP data traffic (type 0) is subdivided into eight subtypes, the latter four of these being acknowledgements for the first four. The first two types (codes 0 and 1) are concerned with the actual transmission of data.

Messages to be sent through the net are sent in one of two ways depending on the size of the message. If the message is less than about 1000 bits then the message is sent as one packet. If the message is larger than 1000 bits then it is

segmented by the IMP into packets of 1000 bits (or less) each. As messages can be up to 8000 bits long, as many as eight packets can be formed from one message.

As a full sized packet occupies a significant amount of buffer space in the IMP, the source IMP first makes sure that the destination IMP has enough buffer space for the message. To this end, the IMP first sends out a request packet that requests space at the destination IMP for the message. When the reply is received the source IMP sends out all eight packets, or as many packets as are necessary. However, if the message is small enough to fit in one packet, no request is sent. The reasoning behind this is that a request message would be as large (approximately) as the data packet itself, so the data is sent as its own request packet.

All messages are acknowledged, not only from IMP to IMP but also from end-to-end. The end-to-end acknowledgement is used as a flow control on the data being sent from the source to the destination. This form of acknowledgement is called a RFNM (Ready for Next Message). Several messages may be outstanding at a time between the source and destination, and in fact the total throughput is often limited by just this factor. Typically, between four and eight messages may be outstanding.

3.2 Host to Host Protocol

Before a user or resource can communicate with another user or resource in the net the two hosts involved have to set up a logical connection. This connection, or link, is given a link number which together with the source and destination host numbers is a unique identifier, but the link number is not necessarily unique in the network. The link number is used to multiplex and demultiplex the traffic travelling between the two hosts.

To establish a connection, the hosts exchange control information over link zero. Link zero is reserved throughout the network for this purpose. The hosts establish the byte size to be used on the connection, the link number that will be used and the processes that are to communicate. This last item is achieved by specifying a source and destination socket number. The hosts must have a priori a common agreement as to what each socket is to represent. For instance socket number three is used to transfer files.

A link only establishes a connection in one direction. Usually of course it is necessary for the hosts to have a connection in both directions when for instance a user is using a remote process interactively. One direction is used to send the lines that the user enters to the process and the other is used to send back messages from the process to the user.

3.3 Analysing Data on a Connection

One of the difficulties of the analysis program is to establish which packets belong to or are the result of which connections. As the reader will have noticed, a source and destination host number and a link number are all that is required to make a connection unique. However, no packets sent between the source and destination contain even this amount of information.

As soon as two hosts begin to transmit traffic to one another, the IMPs try to set up control blocks in the source and destination IMPs to control the traffic. Note that these blocks are set up not for each connection but only for each communicating host pair and that the hosts play no direct part in setting up these blocks.

The blocks are used to keep track of the number of outstanding messages between the source and destination, to generate unique message numbers and to control the buffers required. Should the hosts fall silent for a period of time the IMPs drop the control blocks and try to re-establish them when the hosts become active once more. IMP type 1 packets are used to set up these blocks.

Messages sent from the source host to the destination contain the source and destination IMP numbers, the link number and the destination IMP block number, but not the host numbers. From the block number the destination IMP can look up the host that it is to be delivered to. The reply to the message (RFNM) contains the source and destination IMP numbers, the source IMP block number and the same message number as the message it is acknowledging. Furthermore, the IMP type 1 messages that are used to establish the blocks to be used contain the source and destination IMP numbers, host numbers and block numbers.

As can be seen, the analysis program needs to keep track of several pieces of information in order to attribute a certain

message to a particular connection. The IMP type 1 packets have to be decoded and the information placed in tables. The block number in the data messages is then used to reference this table (together with the source and destination IMP numbers) in order to establish the source and destination Host numbers. The data messages are then placed in a message table to wait for the RFNM. As the RFNM contains the source block number and the data message contained the destination block number, it is not possible to look up the appropriate message directly. The IMP numbers and the source block number are used to reference the block table, in order to establish the destination block number so that the message number may be looked up and the RFNM flagged as having arrived.

Unfortunately the analysis program may not always monitor the opening sequence on a connection. Thus it does not know the source or destination hosts, the source or destination block numbers, or indeed even the byte size being used by the hosts on the connection. The program manages to extract most of this information though from other sources, but the method used to do this will not be explained here as these details are quite complex.

3.4 Uncontrolled Traffic

Although most of the traffic in the network obeys the protocols described above, there is a special type of traffic, usually experimentally, that is not controlled by the flow control mechanism. This type of traffic can be specified by the host and the traffic type is usually referred to as type 3 traffic. (In actual fact, it is subtype 3 of code 0 of IMP type 0 traffic.) There is no flow control on type 3 messages, except of course the buffer capacity of the sending IMP. The messages are not RFNMed and only single packet messages (i.e. less than 1000 bits) are permitted. No attempt is made by the IMPs to deliver these messages in sequence to the destination host, and these messages can be discarded en route if the IMPs run short of buffer space, without informing the source or destination hosts.

3.5 Reserved Link Numbers

Although there are a potential 256 possible numbers, in reality only the first 70 of these are used according to the protocol. Of the rest, most are never used but some are re-

served for experimental purposes. These links are not usually opened or closed by the hosts but are used in a datagram manner across the network. These links are only ever used under specific experimental circumstances. An example of such an experimental link is link number 155 which is reserved for TCP experiments.

4. The Analysis Program

The analysis program analyses the data at three distinct levels corresponding to the levels of protocol existing in the ARPA network. The lowest level is the IMP level and the IMP level routines read the packets from the magnetic tape. If the packet type is determined to be data (typed), then the routines pass the packet up to the host level after first stripping off the IMP protocol layer. Similarly, the host routines strip off the host protocol and interpret any host command messages. If the message type is found to be data then the routines pass information about the data up to the user level. This level is the highest logical level in the program.

Note that only information about the user data is passed to the highest level and not the data itself, as this is never recorded. At the moment the only information available is the data length.

4.1 The Initial Run

The amount of data to be analysed by the program is potentially enormous. In just one hour the program records almost two megabytes of routing data alone. When we consider the number of connections that can exist and the number of bytes of traffic that these will generate, the reader will appreciate that it would not be realisitc to produce reports on every aspect of the recorded information. Furthermore, the line will have been monitored in many cases to determine the performance of one particular connection and thus the user may not be interested in analysis of the other traffic.

To enable the user to select only certain links(and only some particular analysis report of that link) the user must first determine what is actually on the tape. For instance, it may not be enough for the user to know what link is to be analysed, for any connections may have been in existence with

that same link number. Even specifying the hosts may not be sufficient either as the link may have opened and closed several times for different purposes during the data aquisition time.

An inital run is thus made on the tape data to determine the connections that were active. The analysis program determines aninitial run as one that specifies no particular links to be reported on. In an initial run the program scans the tape and produces a table of contents for the tape. This table of contents gives certain basic information about the tape, such as the duration of the data aquisition and the amount of data analysed as well as a list of each connection that was made during the time the measured data was recorded. Each connection is given a separate entry in the table of contents and certain basic information is given about the connection. In particular, the host numbers (if known), the socket numbers, the byte size and the link number are specified as well as the time that the connection began and terminated and the number of messages seen on the link.

The most important aspect of the initial run is that it assigns a unique report number to each connection. The report numbers begin at one and are in ascending sequence. In future runs on this same data, the user must specify connections by this report number. For any connection various reports can be requested by the user. These reports are specified below.

4.2 Overall Reports

Some of the data analysis has nothing to do with individual connections and these reports are specified separately. The most basic report is called the main log.

The main log is a description of each packet that was detected on the line. The report is split into two parts, one for up traffic and one for down traffic. Because of the vast number of packets, certain types of data can be specified to be omitted from the report. The main data types are the four types of IMP traffic that were described in section 3.1, viz., data, control, special and routing. The user can specify which of these types are to be included in the main log. (It is wise not to include routing information, for instance). It is also possible for the user to specify that certain types of traffic are to have the complete packet header dumped in octal as well as having a verbal description. Although the

main log is useful for details of what went up and down the line a possibly more useful report is the Traffic Volume Graph. This is a continuous graph produced on the printer with an entry for each ten seconds that the tapping lasted. Each entry shows not only how much of the line was being used but also subdivides it into three types: routing traffic, other IMP traffic and Host data. It should be emphasized that this volume report is the volume by bytes and not by messages. This report gives a quick indication of the line usage and the overheads involved.

As routing messages seem to occupy such a large percentage of the line capacity and because they indicate how long it should take to communicate with other IMPs on the network, routing packets have three separate reports. Once again these reports are only produced if requested by the user.

The first report is a simple accessibility table. This is a graphic list showing which of the IMPs (currently 67 IMPs are on the network) either of the IMPs on the data aquisition line had access to. In short this is a list of which IMPs in the network were down and when they were down. An entry is produced on this graph for each routing message that is detected.

There is also a Hop/Delay Table available giving a summary of the time it took to communicate with other IMPs in the network. This table gives the average number of hops, the average time in seconds and information on the down time of each IMP. This table is useful when looking at the amount of time it took a particular IMP to RFNM a message on a particular link. It also indicates what effect the satellite had on delay times and how the number of hops is related to the delay.

The final routing report is rather more subtle than the previous two. Although routing messages are sent every .6 seconds this time can vary slightly when the line is heavily loaded. Thus the time between routing messages is an indication of how busy the IMP thinks the line is. The Routing Interval report is a graph that has an entry for each routing message showing the time since the previous routing message. A separate graph is produced for the up and down traffic.

4.3 Connection Reports

For each connection that was monitored there are several reports that can be requested. The first report is a log similar to the main log and the other reports are graphs showing the performance of the connection.

The log report produces a list of the traffic that is directly pertaining to the connection. The report may be requested at a detail level, in which case all the IMP packets to do with the connection are listed, or at the host level in which case only host messages are listed. These two reports may be combined so that both sets of output are produced together. Some other information, such as host control information, that was not sent on the particular connection but has something to do with the commection is also shown on the report. As a further option the user can specify that the packet and/or message headers are to be dumped in octal.

As well as these logs there are five graphs that can be produced. The lengths of these graphs depends upon the duration of the connection; some graphs have an entry for each message that arrives on that link while others have an entry for each unit of time. Each graph is described separately below.

Traffic Volume Graph. This is similar to the volume graph described earlier except that it shows the particular connection in detail. The graph shows the percentage of line capacity being used and breaks down the traffic types into background traffic, IMP overheads and host data. The background traffic consists of all the traffic on the line that is not directly concerned with this link. The IMP overheads indicate the total number of IMP bytes needed to support this link, not of course counting the host data.

Message Arrival Times. This graph has an entry for each message on the link. It shows graphically, on a scale of 10 seconds, the time since the previous message on the link. The times are shown in tenths of a second. For single packet messages the time the message arrives is considered to be the time that the first byte of the packet arrives. For multi-packet messages the arrival time is taken as the time that the first byte of the last packet of the message was detected.

Note that the last packet physically is not necessarily the last packet logically of the message; the packets can have their order changed in flight across the network. However this graph uses the last packet physically of the message. If for some reason the message is retransmitted the program will produce another entry in the graph for the retransmitted message.

Packet Spread Time. This graph indicates how spread out the packets have become while travelling across the net. An entry is made for each complete message that arrives. The graph shows, in increments of .1 seconds up to 10 seconds, the time between the physically first packet of the message and the last packet. This spread time is important because the destination IMP has to reserve buffer space until the complete message has arrived. The IMP cannot deliver the message to the host until all the packets have arrived and the complete message can be reconstructed.

RFNM Time. This graph uses the same scale as the above graph and is used to show the delay between the message being delivered and the RFNM being detected. This graph cannot be produced for type 3 (uncontrolled) messages. The time that the message is delivered is taken to be the time that the first byte of the last packet (physically) of the message was detected. The time that the RFNM was detected is taken as the time that the first byte of the RFNM arrived. It is the RFNM delay that determines the total throughput of the connection.

Message Size. This graph produces an entry for each complete message detected on the connection. It indicates the size of the message up to 1024 bytes. Although for file transfers, where the full message size is almost always used, this graph will not be interesting, it should show how the host deals with interactive users. Bars are written across the graph to indicate the number of packets that were used to transmit the message.

These then are some of the reports that the analysis program will produce. However there are other reports that will be added as more is determined about the network

performance. Already under consideration is the analysis of network control messages (type 1) to give another level of analysis of the IMP to IMP traffic.

5. Some Preliminary Results

Unfortunately, when the analysis program was first devised, we knew very little about the details of IMP to IMP communication. Although there were many areas that we wished to look at we were not sure whether these would be interesting enough to analyse or whether there would be new aspects that we had not considered. For instance, when a message is split up into several packets, do the packets all arrive in quick succession or are they spread out? And if they are spread out, is it a metter of a few tenths of a second or is it many seconds? Is it worth analysing the network control packets (IMP type 1) or are they very rare and very consistent? How does the amount of routing information compare to other traffic? Is it worth analysing the routing details of each routing table or is it very consistent with minimum variations? How much special traffic usually goes through the net? Is there enough to analyse and what are its important characteristics?

It became obvious that we needed a way to inspect the traffic being sent from IMP to IMP before the analysis program could be completed. It was decided to write the analysis program in two phases. The first phase would merely produce the main log as described earlier. When the important parameters and aspects of IMP to IMP communication were determined, the second pahse would be started to complete the analysis program. At the present time, phase one is complete and phase two is nearing completion.

5.1 Phase One

The phase one program will produce a detail listing of all the traffic noted on the IMP communication lines. Because of the large amount of data, there is a certain amount of flexibility in the program to allow certain traffic types to be suppressed. For instance, due to the predominance of routing traffic it is usually a good idea to suppress the printing of this traffic when inspecting, say, data traffic.

The program will also perform a small amount of host traffic analysis. This consists of decoding the host header at the beginning of a host message, and if the host message is a control message, it will also decode each individual message that the hosts are sending. Once again this is optional, and the user may suppress this function of the phase one program.

5.2 What We Learned

The phase one program brought out many problems associated with the recording of information and in analysing the results. These problems were caused mainly by small details which under normal circumstances would be of no interest to anyone except the implementers of the IMP software or hardware. For instance, it was known that the IMPs send data along the communication lines so that each pair of bytes is reversed (i.e. bytes 0,1, 2 and 3 would be sent in order, 1,0,3,2), but it was not known that the sumcheck at the end of the packet does not do this, but is sent in the normal fashion.

The most interesting results were in the listing of the traffic types. Of the four main categories of IMP traffic (data, network control, routing and special) we found that even when there is a significant amount of data traffic, the routing messages still account for over 60% of the traffic sent from London to Norway. Furthermore, in all the analysis done so far (several hours worth) there have been no special packet types (type 3) at all.

A disturbing property of the London to Norway line that we discovered, was the relatively high level of retransmissions that takes place. The London IMP will retransmit a message to Norway if it has not received an acknowledgement from Norway in 2.1 seconds, it appears. The only reason that Norway would not acknowledge a message is if (a) the message was corrupted between here and Norway (sumcheck does not agree) or (b) the IMP in Norway is so busy or has so many buffers full that it cannot accept any more traffic. When we looked into the flow of traffic in detail it seems that it is the latter that causes the problem.

At Norway much data is injected into the Arpanet destined for the U.S.A. When London traffic is added to this there is a potential bottleneck at Norway. Hence, if we send data rapidly enough to the U.S.A. Norway quickly becomes swamped, and rejects all packets until it has freed some of the buffers.

This analysis helped to explain some problems being encountered with TCP experiments that were being performed between our installation and California. An important part of the experiment consisted of an analysis of windowing, a flow control mechansim. It was found that when the window was enlarged past a certain threshold the throughput was far less than expected. The phase one program helped to establish the real cause of this and showed that it was not a direct fault of the TCP protocols or the source and destination host machines.

We have also noticed some hitherto unforseen repercussions of the present host to host protocol. The protocol suggests that when a host comes up on the network after being down for any reason, it should send "resets", a special host control message, to all hosts on the network to indicate that any connections that were in progress when the machine crashed are to be cancelled. Thus in a short space of time, the host might try to send up to 256 reset messages into the network.

However, the IMP will not allow communication with another host in the network unless a pair of network control messages have been exchanged between itself and the destination IMP. Thus for each reset message the IMP must exchange a pair of messages with an IMP on the network. This effectively triples the number of messages that the host sent into the network. Furthermore, the Norway IMP quickly swamps and the London IMP starts retransmissions. All other traffic travelling through our IMP is affected too. To make matters worse, approximately 2½ minutes later, the IMP determines that there has been no more traffic on all those connections, and so the IMP tries to reset all the connections it set up in order to send the one host reset message. This IMP resetting again requires the exchange of two packets, and the line is once again flooded.

6. Conclusions

From the analysis we have performed so far a great deal has been learned about the performance of the ARPA network. The second phase of the analysis program is almost complete now and it is hoped that this will give even more insight into the details of host to host communication. There are some things that are very difficult to do from just the listing of the IMP to IMP traffic, such as the mean time between messages for a file transfer that required several thousand messages. As indicated in this paper, the analysis program will produce results such as these. But this will only be the starting point. It will then be necessary to understand what is happening in the network and to be able to explain it in order to be able to make suggestions for improvements in other future networks.

MESSAGE ACKNOWLEDGEMENT DELAYS

The following two graphs illustrate acknowledgement delays (RFNM delays) experienced for a particular connection on the ARPA network. The traffic was being sent from London to Stanford in California.

- The first graph is part of a continuous graph in which each RFNM is represented as one bar line on the graph. To the left of each bar is the message number and the delay time from the message to the RFNM. The bar illustrates this time graphically.

- The second graph is a summary chart, indicating the distribution of RFNM times. For instance it can be seen that approximately 20% of the traffic had a RFNM delay of between 1.36 and 1.44 seconds. Furthermore, no acknowledgement was received in less than 1.12 seconds, and no acknowledgement took more than 2.00 seconds.

MESSAGE/RFNM DELAY TIME (ALL TIMES IN SECS)

```
MSG     TIME          0.40       0.80       1.20       1.60       2.00       2.40
  1     1.71  ******************************************
  2     1.67  *****************************************
  3     1.21  ******************************
  4     1.33  *********************************
  5     1.88  ***********************************************
  6     1.83  *********************************************
  7     1.52  **************************************
  8     1.37  **********************************
  9     1.82  *********************************************
 10     1.42  ***********************************
 11     1.67  *****************************************
 12     1.77  ********************************************
 13     1.55  **************************************
 14     1.36  **********************************
 15     1.87  **********************************************
 16     1.74  *******************************************
 17     1.46  ************************************
 18     1.26  *******************************
 19     1.22  ******************************
 20     1.38  **********************************
 21     1.18  *****************************
 22     1.70  ******************************************
 23     1.67  *****************************************
 24     1.66  *****************************************
 25     1.81  *********************************************
 26     1.39  **********************************
 27     1.74  *******************************************
 28     1.34  *********************************
 29     1.73  *******************************************
 30     1.77  ********************************************
 31     1.64  *****************************************
 32     1.41  ***********************************
 33     1.36  **********************************
 34     1.25  *******************************
 35     1.85  **********************************************
 36     1.97  *************************************************
 37     1.50  *************************************
 38     1.40  ***********************************
 39     1.68  ******************************************
 40     1.87  **********************************************
 41     1.82  *********************************************
```

RFNM DELAY DISTRIBUTION

MAXTIME	NUMBER	10%	20%	30%	40%	50%
0.08	0					
0.16	0					
0.24	0					
0.32	0					
0.40	0					
0.48	0					
0.56	0					
0.64	0					
0.72	0					
0.80	0					
0.88	0					
0.96	0					
1.04	0					
1.12	0					
1.20	1	**				
1.28	4	*********				
1.36	2	****				
1.44	8	**********				
1.52	2	****				
1.60	2	****				
1.68	5	**********				
1.76	6	**********				
1.84	6	**********				
1.92	4	*********				
2.00	1	**				
2.08	0					
2.16	0					
2.24	0					
2.32	0					
2.40	0					
2.48	0					
2.56	0					
2.64	0					
2.72	0					
2.80	0					
2.88	0					
2.96	0					
3.04	0					
3.12	0					
3.20	0					
3.28	0					
3.36	0					
3.44	0					
3.52	0					
3.60	0					
3.68	0					
3.76	0					
3.84	0					
3.92	0					
4.00	0					

SAMPLE REPORT OUTPUT

Direction of packet

Date and time that monitoring began

```
ANALYSIS REPORT OF TAP DATA TAKEN ON 15/07/76 AT 20.26.27                    PAGE 4

  UP 20:26:47.21    IMP: MSG 188, RFNM, FROM IMP 42, TO IMP 5, TRANS BLK 36, TRANS USE 2, PRIORITY

DOWN 20:26:47.84    IMP: MSG 1, DATA, FROM IMP 9, TO IMP 42, RCV BLK 4, RCV USE 5, SINGLE PKT MSG, PRIORITY, 10 BYTES OF TEXT
                    HOST:LINK 0 (CONTROL), BYTE SIZE 8, BYTE COUNT 2
                    HOST:ECO, DATA 10101010

  UP 20:26:47.89    IMP: MSG 1, RFNM, FROM IMP 42, TO IMP 9, TRANS BLK 4, TRANS USE 3, PRIORITY

  UP 20:26:47.96    IMP: MSG 1, DATA, FROM IMP 42, TO IMP 5, RCV BLK 3, RCV USE 5, SINGLE PKT MSG, PRIORITY, 10 BYTES OF TEXT
                    HOST:LINK 0 (CONTROL), BYTE SIZE 8, BYTE COUNT 3
                    HOST:ERP, DATA 10101010

DOWN 20:26:48.55    IMP: MSG 4, RFNM, FROM IMP 56, TO IMP 42, TRANS BLK 4, TRANS USE 10, PRIORITY

DOWN 20:26:48.57    IMP: MSG 169, RFNM, FROM IMP 5, TO IMP 42, TRANS BLK 1, TRANS USE 10, PRIORITY

DOWN 20:26:48.60    IMP: MSG 5, RFNM, FROM IMP 56, TO IMP 42, TRANS BLK 4, TRANS USE 10, PRIORITY

  UP 20:26:48.62    IMP: MSG 170, DATA, FROM IMP 42, TO IMP 5, RCV BLK 14, RCV USE 3, SINGLE PKT MSG, PRIORITY, 10 BYTES OF TEXT
                    HOST:LINK 2, BYTE SIZE 8, BYTE COUNT 3

DOWN 20:26:48.63    IMP: MSG 169, DATA, FROM IMP 5, TO IMP 42, RCV BLK 3, RCV USE 6, SINGLE PKT MSG, PRIORITY, 16 BYTES OF TEXT
                    HOST:LINK 0 (CONTROL), BYTE SIZE 8, BYTE COUNT 8
                    HOST:ALL, LINK 2, MSG SPACE 2, BIT SPACE 360

  UP 20:26:48.69    IMP: MSG 189, RFNM, FROM IMP 42, TO IMP 5, TRANS BLK 36, TRANS USE 2, PRIORITY

DOWN 20:26:48.70    IMP: MSG 4, DATA FROM IMP 56, TO IMP 42, RCV BLK 5, RCV USE 6, SINGLE PKT MSG, 54 BYTES OF TEXT
                    HOST:LINK 155
```

Host analysis

Description of each IMP packet

Time of packet in hours:minutes:seconds.hundredths

EXAMPLE 1

This report shows the raw IMP packets that were monitored. NULL packets are a type of routing packet, and the reader will note the predominance of routing traffic.


```
DEFAULT OPTIONS ARE:
      DON'T PRINT   TYPE 0: DATA
      DON'T PRINT   TYPE 1: NETWORK CONTROL
      DON'T PRINT   TYPE 2: SWITCH CONTROL
      DON'T PRINT   TYPE 3: SPECIAL

OPTIONS PARM WAS :DATA,CONT,ROUT,SPEC

YOU HAVE SELECTED:
      PRINT   TYPE 0: DATA
      PRINT   TYPE 1: NETWORK CONTROL
      PRINT   TYPE 2: SWITCH CONTROL
      PRINT   TYPE 3: SPECIAL
```

```
DOWN 20:26:34.01    IMP: NULL PKT, I AM A STUB, SRC IMP 41
DOWN 20:26:34.02    IMP: ROUTING TABLE, SRC IMP 41
  UP 20:26:34.10    IMP: NULL PKT, I AM A STUB, SRC IMP 42
  UP 20:26:34.12    IMP: ROUTING TABLE, SRC IMP 42
  UP 20:26:34.74    IMP: NULL PKT, I AM A STUB, SRC IMP 42
  UP 20:26:34.76    IMP: ROUTING TABLE, SRC IMP 42
DOWN 20:26:35.29    IMP: NULL PKT, I AM A STUB, SRC IMP 41
DOWN 20:26:35.30    IMP: ROUTING TABLE, SRC IMP 41
  UP 20:26:35.38    IMP: NULL PKT, I HEARD YOU, I AM A STUB, SRC IMP 42
  UP 20:26:35.40    IMP: ROUTING TABLE, SRC IMP 42
DOWN 20:26:35.93    IMP: NULL PKT, I HEARD YOU, I AM A STUB, SRC IMP 41
DOWN 20:26:35.95    IMP: ROUTING TABLE, SRC IMP 41
  UP 20:26:36.02    IMP: NULL PKT, I AM A STUB, SRC IMP 42
  UP 20:26:36.04    IMP: ROUTING TABLE, SRC IMP 42
DOWN 20:26:36.57    IMP: NULL PKT, I AM A STUB, SRC IMP 41
DOWN 20:26:36.59    IMP: ROUTING TABLE, SRC IMP 41
  UP 20:26:36.66    IMP: NULL PKT, I AM A STUB, SRC IMP 42
  UP 20:26:36.68    IMP: ROUTING TABLE, SRC IMP 42
DOWN 20:26:37.21    IMP: NULL PKT, I AM A STUB, SRC IMP 41
DOWN 20:26:37.23    IMP: ROUTING TABLE, SRC IMP 41
  UP 20:26:37.30    IMP: NULL PKT, I AM A STUB, SRC IMP 42
  UP 20:26:37.32    IMP: ROUTING TABLE, SRC IMP 42
  UP 20:26:37.94    IMP: NULL PKT, I AM A STUB, SRC IMP 42
  UP 20:26:37.96    IMP: ROUTING TABLE, SRC IMP 42
  UP 20:26:38.44    IMP: GET BLOCK, FROM IMP 42, TO IMP 56, SRC BLK 4, SRC USE 10, SRC HOST 1, DEST HOST C, PRIORITY
DOWN 20:26:38.49    IMP: NULL PKT, I AM A STUB, SRC IMP 41
DOWN 20:26:38.51    IMP: ROUTING TABLE, SRC IMP 41
  UP 20:26:38.59    IMP: NULL PKT, I HEARD YOU, I AM A STUB, SRC IMP 42
```

UP 20:26:38.60 IMP: ROUTING TABLE, SRC IMP 42

DOWN 20:26:38.63 IMP: NULL PKT, I AM A STUB, SRC IMP 41

DOWN 20:26:39.13 IMP: NULL PKT, I HEARD YOU, I AM A STUB, SRC IMP 41

DOWN 20:26:39.15 IMP: ROUTING TABLE, SRC IMP 41

UP 20:26:39.23 IMP: NULL PKT, I AM A STUB, SRC IMP 42

UP 20:26:39.24 IMP: ROUTING TABLE, SRC IMP 42

UP 20:26:39.74 IMP: UNCONTROLLED PACKET (TYPE 3), FROM IMP 60 HOST 255, TO IMP 40 HOST 0, 68 BYTES OF TEXT

DOWN 20:26:39.77 IMP: NULL PKT, I AM A STUB, SRC IMP 41

DOWN 20:26:39.79 IMP: ROUTING TABLE, SRC IMP 41

UP 20:26:39.87 IMP: NULL PKT, I AM A STUB, SRC IMP 42

UP 20:26:39.88 IMP: ROUTING TABLE, SRC IMP 42

DOWN 20:26:39.91 IMP: GET BLOCK RPLY, FROM IMP 56, TO IMP 42, SRC BLK 5, SRC USE 13, DEST BLK 4, DEST USE 10, PRIORITY

UP 20:26:40.00 IMP: UNCONTROLLED PACKET (TYPE 3), FROM IMP 60 HOST 255, TO IMP 40 HOST 0, 114 BYTES OF TEXT

UP 20:26:40.12 IMP: MSG 0, DATA, FROM IMP 42, TO IMP 56, RCV BLK 5, RCV USE 13, SINGLE PKT MSG, 56 BYTES OF TEXT

DOWN 20:26:40.18 IMP: NULL PKT, I AM A STUB, SRC IMP 41

UP 20:26:40.19 IMP: MSG 1, DATA, FROM IMP 42, TO IMP 56, RCV BLK 5, RCV USE 13, SINGLE PKT MSG, 56 BYTES OF TEXT

DOWN 20:26:40.24 IMP: NULL PKT, I AM A STUB, SRC IMP 41

DOWN 20:26:40.31 IMP: NULL PKT, I AM A STUB, SRC IMP 41

DOWN 20:26:40.41 IMP: NULL PKT, I AM A STUB, SRC IMP 41

DOWN 20:26:40.43 IMP: ROUTING TABLE, SRC IMP 41

UP 20:26:40.51 IMP: NULL PKT, I AM A STUB, SRC IMP 42

UP 20:26:40.52 IMP: ROUTING TABLE, SRC IMP 42

DOWN 20:26:41.06 IMP: NULL PKT, I AM A STUB, SRC IMP 41

DOWN 20:26:41.07 IMP: ROUTING TABLE, SRC IMP 41

UP 20:26:41.15 IMP: NULL PKT, I AM A STUB, SRC IMP 42

UP 20:26:41.16 IMP: ROUTING TABLE, SRC IMP 42

DOWN 20:26:41.70 IMP: NULL PKT, I AM A STUB, SRC IMP 41

DOWN 20:26:41.71 IMP: ROUTING TABLE, SRC IMP 41

UP 20:26:41.79 IMP: NULL PKT, I HEARD YOU, I AM A STUB, SRC IMP 42